S A D T

Structured Analysis and Design Technique

Défini par Douglas T. ROSS, en 1972

Normalisé pour le compte de l'US Air Force et l'ICAM (Integrated Computer Aided Manufacturing) pour favoriser la coopération entre les industriels de l'aéronautique

➔ IDEF0, IDEF1

site : www.idef.com

Diffusé à partir de 77,
Assez largement utilisé jusque en 90,
Un formalisme de référence pour l'approche fonctionnelle du logiciel.

Motivations, dans le contexte des années 70

- Permettre de définir de façon précise et rigoureuse les *fonctions* que doit assurer un système
- Organiser la coopération entre les parties prenantes :
  A Language for communicating Ideas.

SADT =

Un formalisme pour exprimer la définition des fonctions d'un système (SA)
=> modèles

+

Une démarche, mode d'emploi pour utiliser au mieux ce formalisme (DT)
=> processus d'élaboration de modèles

Plan

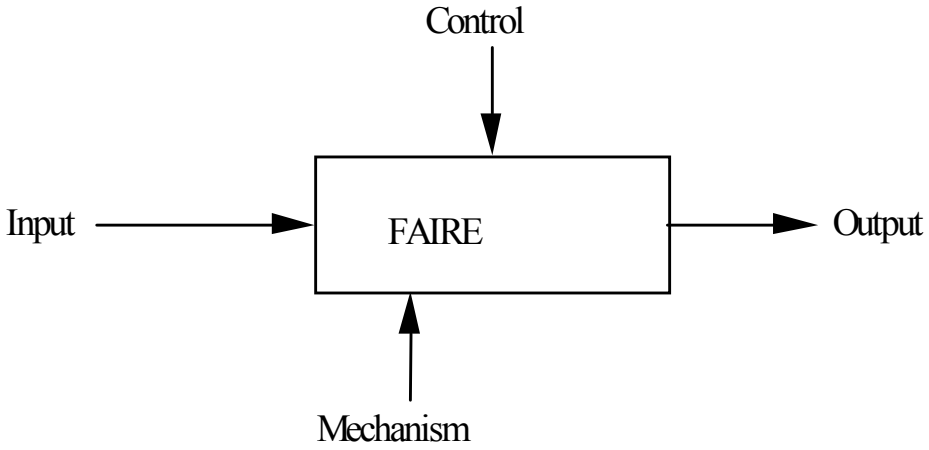
Les principes 2
Les modèles SADT 10
Le processus de modélisation de SADT 18

Les Principes

Les objectifs, et les procédés utilisés pour les atteindre

Domaine d'application

Un système
est capable de faire qq chose
en produisant un résultat
grâce à ce qui lui est fournit par son environnement



Tout système, comportant un assortiment d'acteurs humains, d'automatismes et de logiciels, ou plus généralement une tâche à réaliser :
systèmes industriels
système d'information
résolution de problème, "comment faire pour …"

Modéliser

Etablir une description, une représentation du système.
« Pour un opérateur O, un objet M est un modèle d'un objet S si O peut utiliser M pour répondre à des questions Q qu'il se pose au sujet de S » (M. Minsky).

Tout modèle M dépend de
Le **contexte** : délimitation de l'objet d'intérêt (quel objet S ?)
Le **point de vue** du modélisateur (quel opérateur O ?)
L'**objectif** (quelles questions Q ?)

Le contexte

Un système est délimité par son interface avec son environnement :
tout échange se produit entre un élément **dans** le système et un élément **hors** du système.

Le contexte

Un système est délimité par son interface avec son environnement :
tout échange se produit entre un élément **dans** le système et un élément **hors** du système.

Le point de vue du modélisateur

Aucun modèle ne peut tout dire sur un système.
L'importance relative des éléments du système à prendre en compte dépend de la perspective adoptée, la responsabilité, le rôle du modélisateur vis à vis du système
exple :

Utilisateurs finaux,
Opérateur d'exploitation,
Mainteneur, Installateur
Conformité aux réglementations applicables
Commercial

L'objectif d'une modélisation (purpose)

Comprendre un système existant pour :

Analyser l'origine des dysfonctionnements
Le réorganiser
Expliquer comment il fonctionne, formation des utilisateurs, responsables

Décrire un système futur pour :

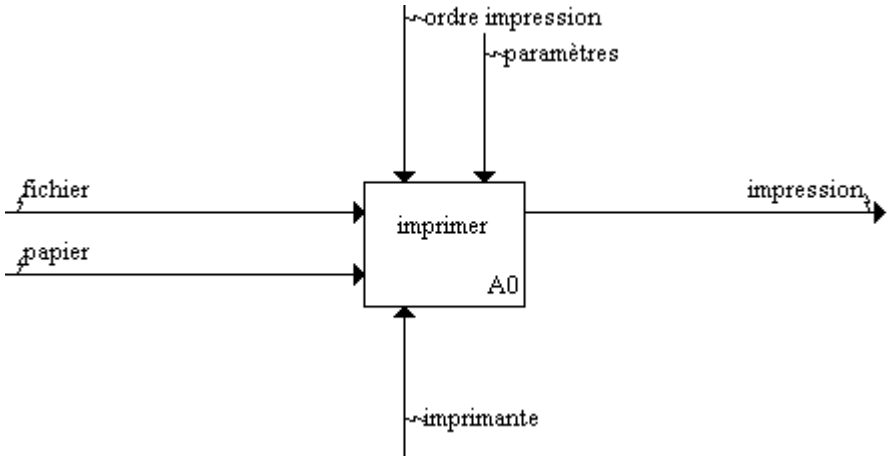
Exprimer les besoins : que doit faire le système ?
Définir ses relations avec son environnement
Planifier une activité : quelles sont les tâches à réaliser

L'objectif fournit le critère permettant de décider entre +sieurs alternatives,
notamment de l'arrêt du processus de modélisation.

=> SADT ne s'utilise pas pour décrire la structure d'un système, quels sont ses constituants, …
mais, pendant les phases amonts d'un projet Logiciel, pour comprendre

ce que fait ce système, les besoins auxquels il doit répondre
quelles opérations doivent être appliquer aux entrées pour produire les sorties.

Sous forme d'un ensemble de contraintes

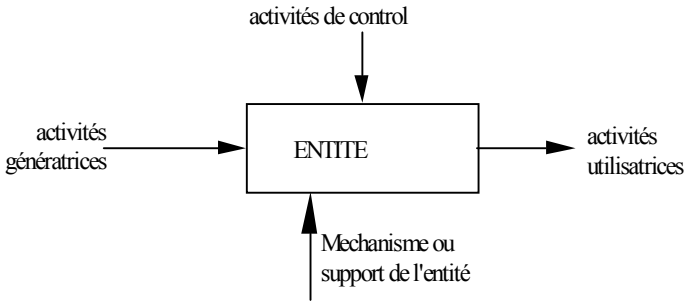


La dualité activité / entité

Activités : ce que fait le système, les opérations, traitements qu'il réalise
Entités, données : ce que le système manipule, utilise, produit
Activités et données
sont indispensables
prennent leur sens l'un par l'autre.

Modèles d'activité et modèles de données sont deux visions du système,
leur confrontation permet de les valider l'un par l'autre.

Modélisation des activités -> actigrammes
des données -> datagrammes



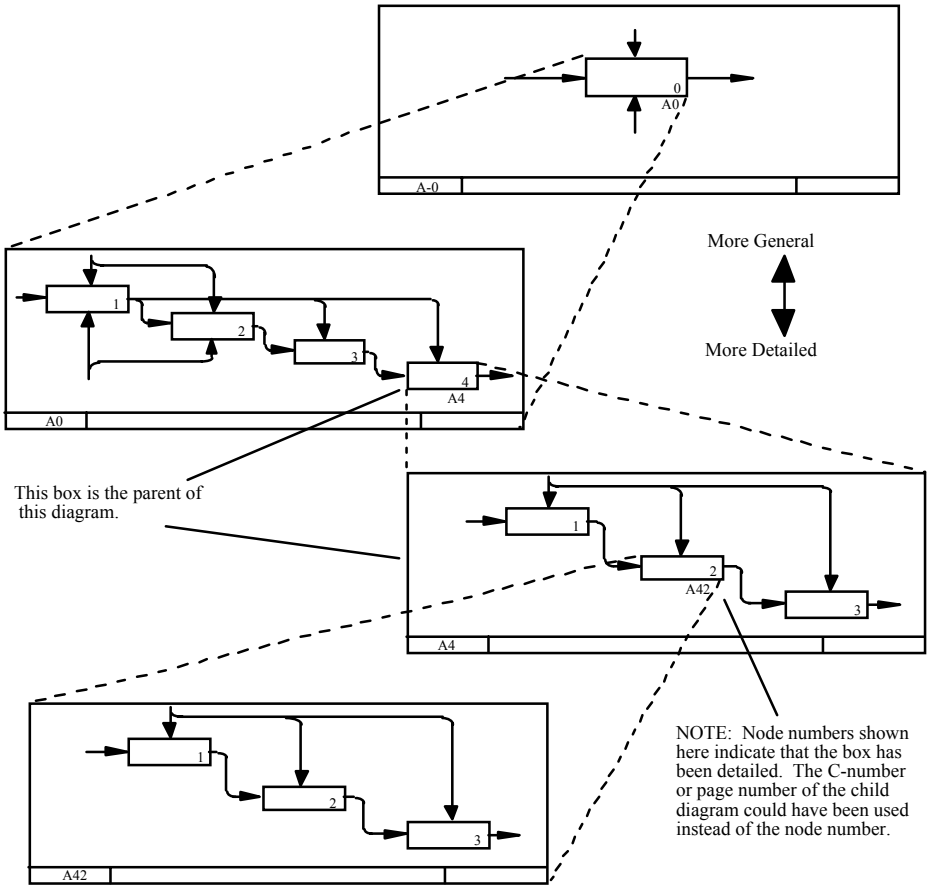
En pratique, on n'utilise pas les datagrammes SADT mais le formalisme E-A.

Structure hiérarchique du modèle et du système

Selon le principe cartésien, tout système peut être décomposé en (sous-)systèmes,
qui à leur tour peuvent être décomposés …

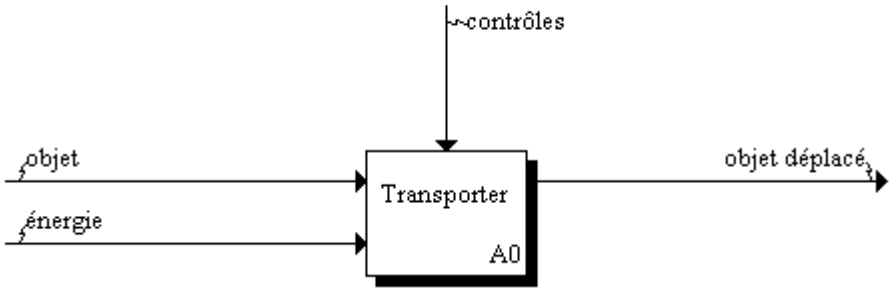
Les **arbres** sont des structures très simples,
il suffit de définir, pour chaque nœud, ses relations avec
son parent,
ses "frères".

niveau i – 1 : **pourquoi**, le contexte du niveau i
niveau i : **quoi**
niveau i + 1 : **comment** du niveau i

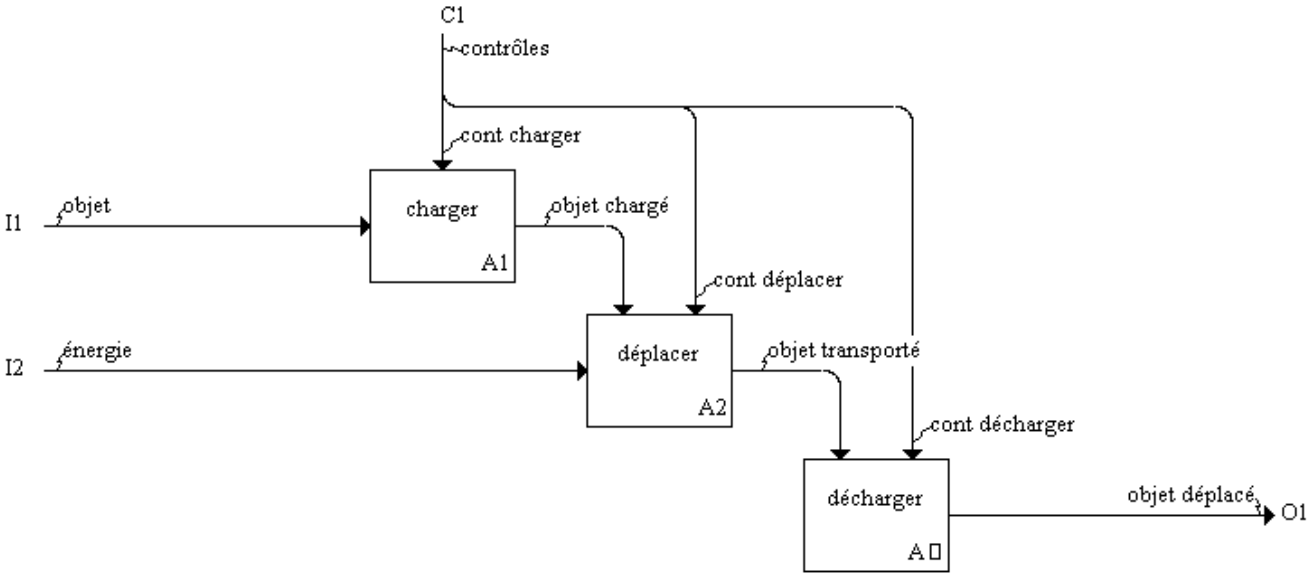


Cette structure hiérarchique du système détermine :

- La structure du modèle, sous forme d'une hiérarchie de diagrammes,
- L'organisation de l'ensemble des fonctions que le système doit réaliser.
- La structure du processus de modélisation, selon une analyse descendante.



Modèle SADT d'un dispositif de transport



L'organisation du travail de modélisation qui fait quoi quand

Approche pragmatique, qui vise à l'efficacité

Bien découper les tâches, pour qu'elles soient le plus indépendantes possible.
Définir clairement le rôle et les responsabilités de chacun.

Définir clairement l'ordonnancement des tâches :
==> Le cycle auteur - lecteur

Pour chaque tâche, indiquer comment procéder (pragmatique du formalisme).

Utiliser des moyens de communication efficace
=> document écrit, de structure normalisée.

L'intelligibilité des modèles

Un modèle est un support pour l'élaboration et le partage de connaissances
Elaboration et partage sont indissociables, car un modèle valide est l'expression d'un consensus.

L'effectivité de l'évaluation d'un modèle par une personne dépend de sa compréhension du modèle.
=> Le modèle doit être bien structuré,
pour rendre facile l'accès à tout niveau de détail.

=> Chaque élément du modèle doit être aisément intelligible,
exprimer ce que l'on peut dire, en 1 page, d'une activité du système.

Un modèle doit comporter

1. une description du système considéré,
2. une justification des choix de modélisation,
   en réponse aux questions que l'on peut se poser sur la description,
3. la trace du processus de son élaboration.

Graphique + texte

Graphique : pour décrire, selon la techniques des plans d'architecte et du dessin industriel
concis, rigoureux et précis, favorisent le cohérence,
avec un grand nombre de convention, pour faciliter la compréhension
Texte : pour le commentaire explicatif

- les éléments du modèles qui ne se prêtent pas à une représentation graphique,
- informations **sur** le modèle, qui indiquent comment l'interpréter.

Structure du modèle d'un Ascenseur :

A0: ascenseur
- A1: calculer destination
  - A11: mémoriser appel
  - A12: mémoriser étage demandé
  - A13: calculer
- A2: déplacer cabine
- A3: actionner portes
  - A31: ouvrir
  - A32: attendre
  - A33: vérifier charge
  - A34: fermer

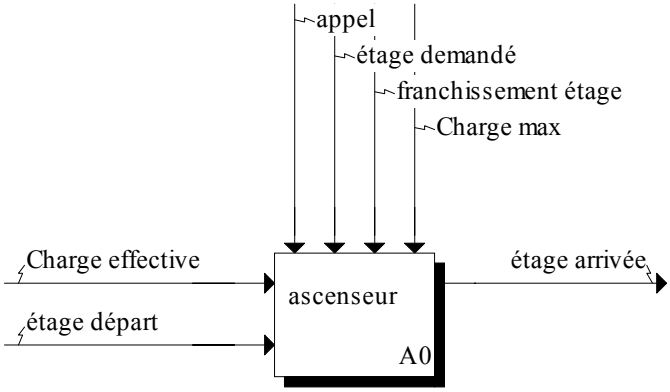


Diagramme A-0

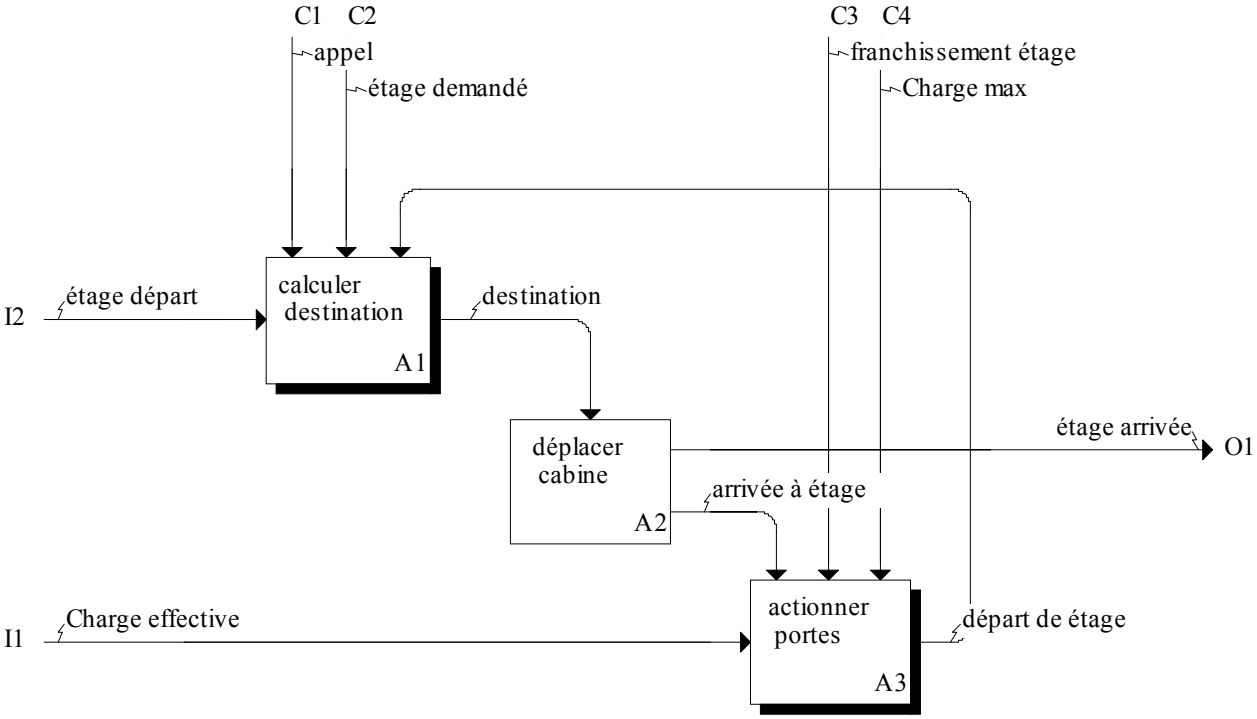


Diagramme A0

Legend:

I = Input
C = Control
O = Output
M = Mechanism
Concept
Unused Concept
Activity
Activity with no Controls
D = Decomposition
A = Call Arrow

Activities	Concepts	appel	arrivée à étage	Charge effective	Charge max	destination	départ de étage	franchissement étage	étage arrivée	étage demandé	étage départ
		C		I	C			C	O	C	I
A1: calculer destination	D	C				O	C			C	I
A2: déplacer cabine			O			C			O		
A3: actionner portes	D		C	I	C		O	C			

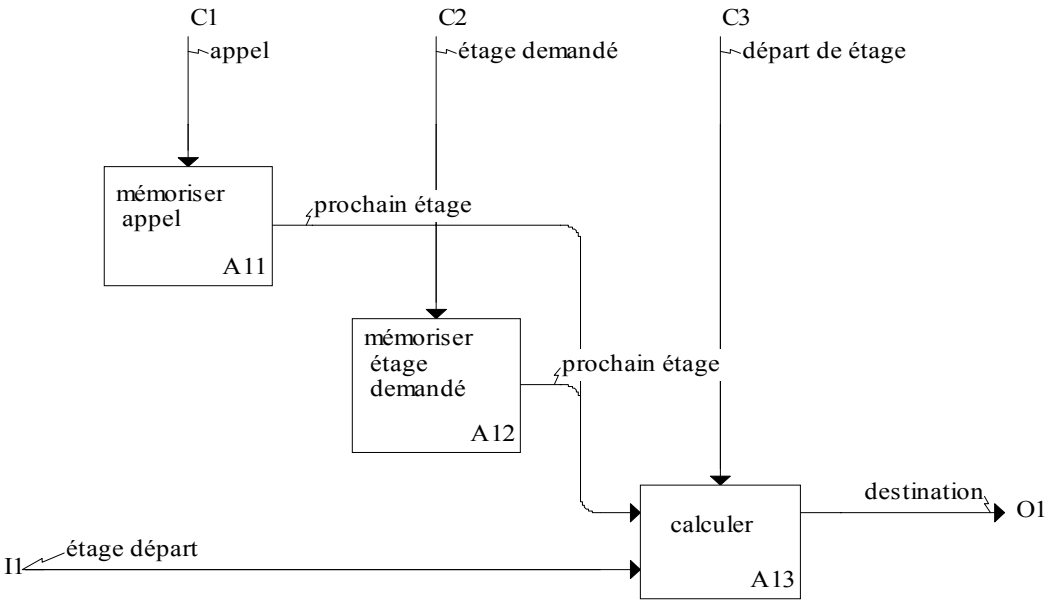


Diagramme A1

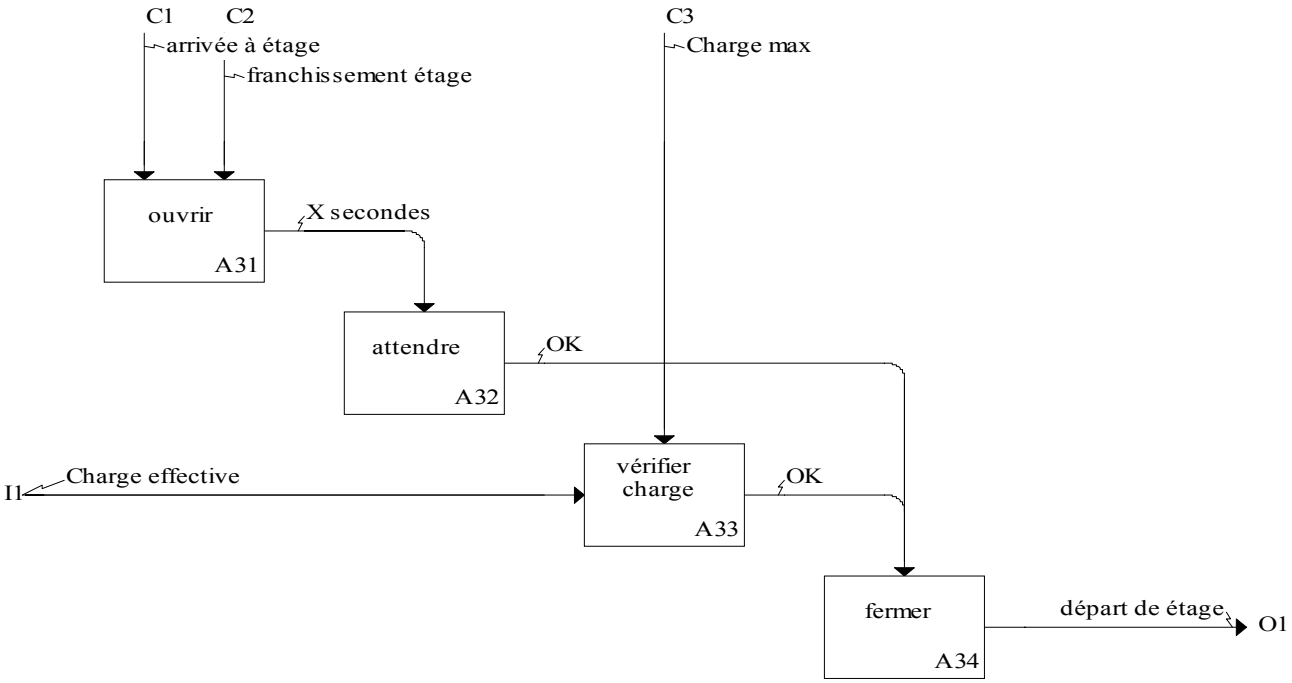
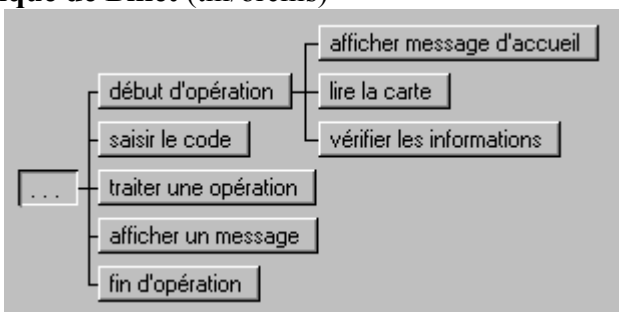


Diagramme A3

Un Distributeur Automatique de Billet (tni/orchis)



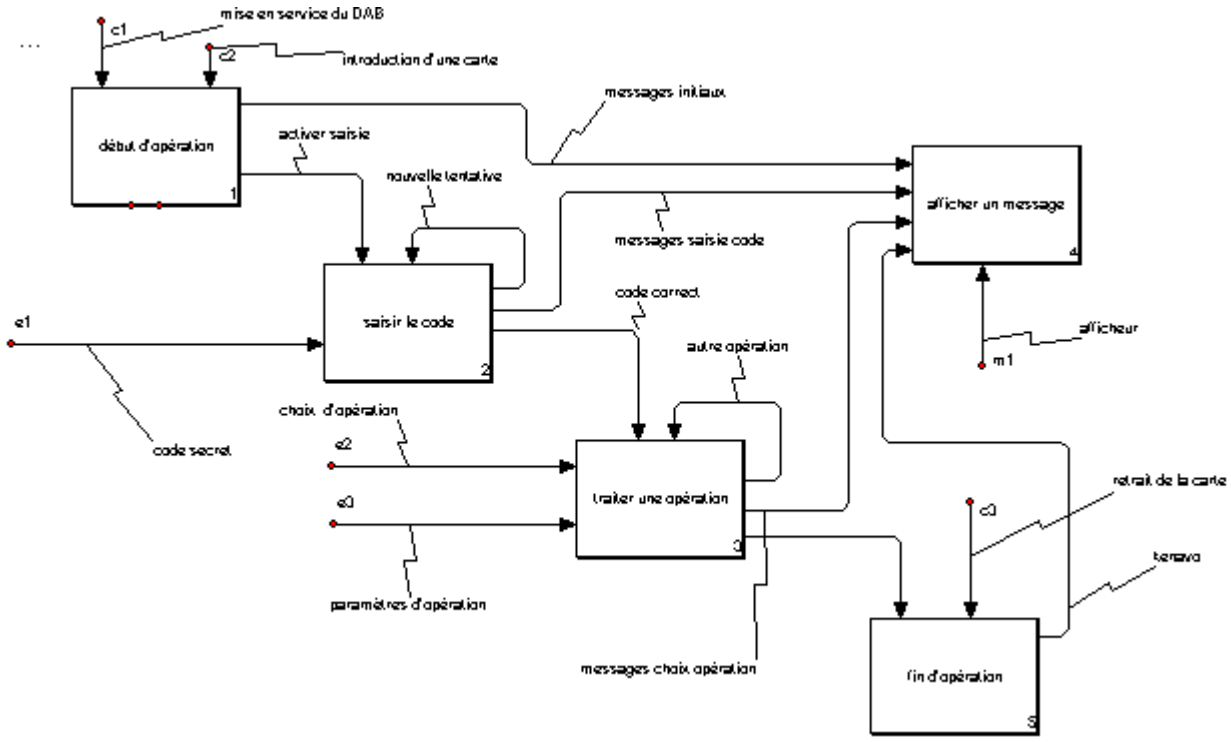


Diagramme A0

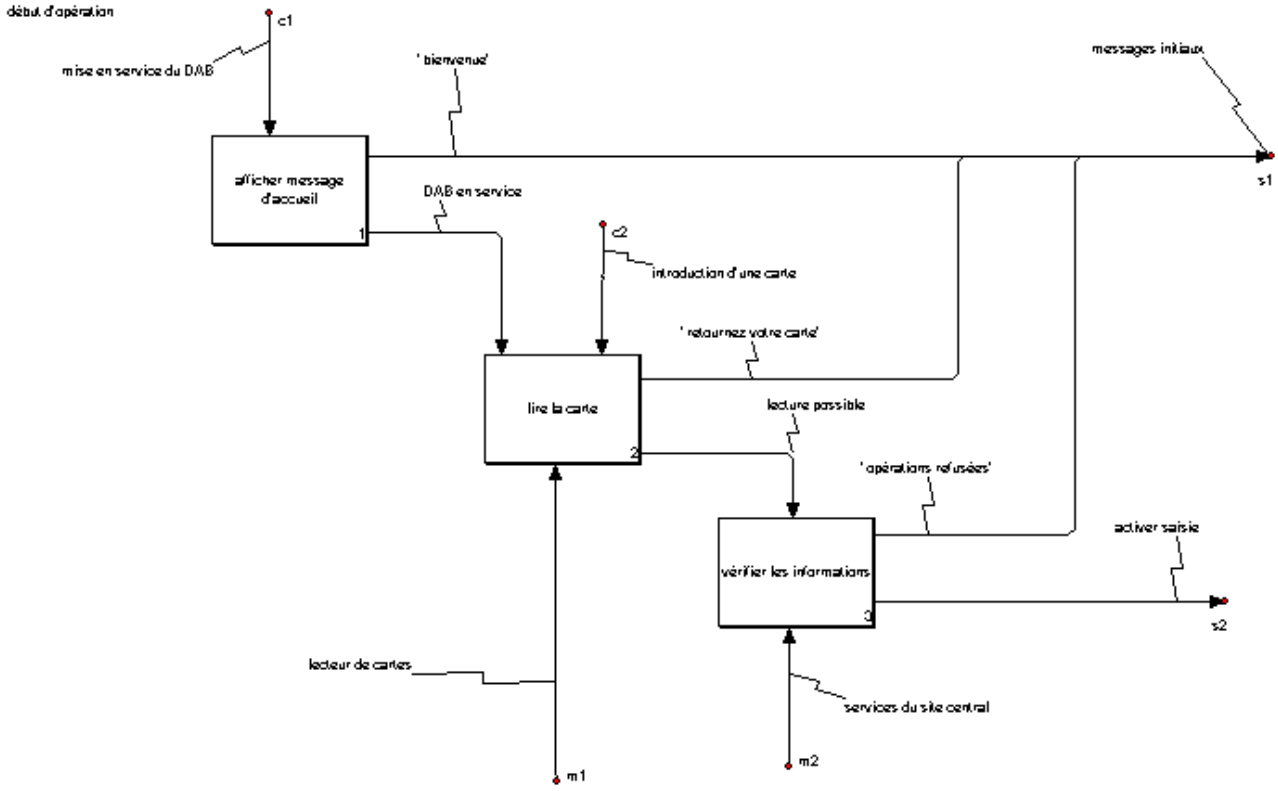


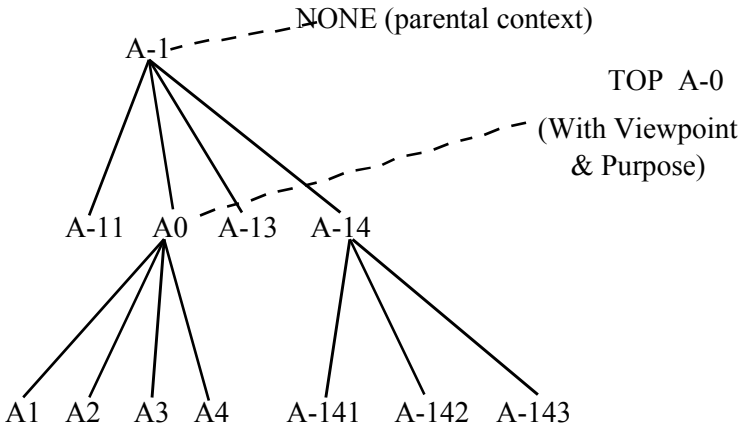
Diagramme A1

Les Modèles SADT

Un modèle est caractérisé par :
son contexte : le système, l'activité, ... considéré
l'objectif,
le point de vue.
Dans le cadre d'un même projet, on peut avoir plusieurs modèles.

La structure d'un modèle

- énoncé de l'objectif et point de vue du modèle
- une arborescence de diagrammes,
  la racine A-0 : le diagramme de contexte (comporte une seule activité)
  chaque nœud est identifié en fonction de sa position
  A0 : la 1ère décomposition du système (seul fils de A-0)
- éventuellement, nœud A-1 qui décrit le système englobant, l'environnement de A-0
- un glossaire des termes spécifiques utilisés



La structure d'un nœud

Chaque nœud décrit une fonction du système
- dans son contexte (fournit par son nœud père),
- à un certain niveau de détail

- un diagramme (actigramme) qui décrit la fonction en terme de sous-fonctions
- un ensemble de notes (texte, ou schéma PES (Pour Explication Seulement)),
  identifiées par un numéro
  explications : éléments de la description qui ne peuvent être exprimés par le diagramme **|n|**
  commentaires : justification, alternatives **(n)**
- éventuellement, un glossaire locale.

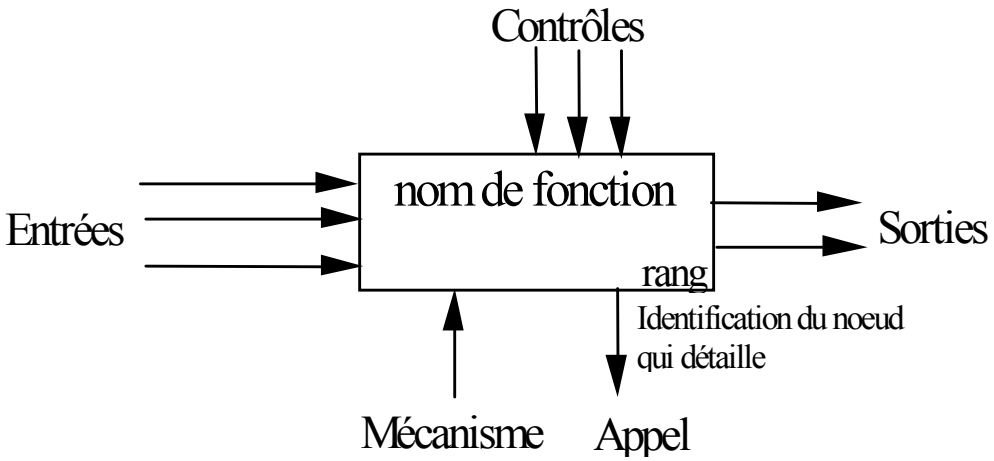
Chaque nœud peut être vu comme la racine d'une sous-arborescence,
et donc un modèle d'un sous-système

La structure d'un diagramme

- un ensemble de boîtes, correspondant aux sous-fonctions
  - au moins 3 (assez d'information)
  - au plus 6 (mais pas trop)
- un ensembles de flèches, correspondant aux données/objets manipulés
- des point d'ancrage boîte-flèche, indiquant les relations
  - entre les boîtes
  - entre des loîtes et l'environnement.

Les boîtes

Chaque boîte correspond à une (sous-)fonction.
nom de la fonction : un verbe, ou une expression verbale, aussi spécifique que possible
numéro d'ordre : entier (1 ... 6), qui identifie la boîte dans le diagramme
Identificateur du noeud fils qui détaille la boîte : Diagram Reference Expression



L'interface d'une boîte :

- **sorties** : résultat fourni à chaque exécution de l'opération.
- **entrées** : les données/objets auxquels l'opération s'applique, modifiés par son exécution
- **contrôles** : événement déclenchant, indication sur les modalités d'exécution
- **mécanismes** : ressources ou dispositif utilisés, nécessaires pour pouvoir réaliser l'opération
  - QUI fait l'opération, BD accédée, ...
- **appel** : sous-système utilisé de façon non exclusive
  - (ne peut donc pas être détaillé comme une ss-fonction)

Toute boîte a :
- au moins 1 contrôle
- au moins 1 sortie
- au plus 10 flèches connectées.

Les Entrées et Mécanismes sont des contraintes qui conditionnent l'exécution de l'opération,
- ce dont elle a besoin pour pouvoir s'exécuter;

Ceux sont les Contrôles qui déterminent **Quand** l'opération s'exécute.

Une exécution de l'opération
- n'a pas nécessairement besoin de chacun des intrants
- produit au moins une sortie, mais pas nécessairement chacune des sorties

modalité d'activation d'une boite, donnée en note : I2, C1, C3 → S1, S2

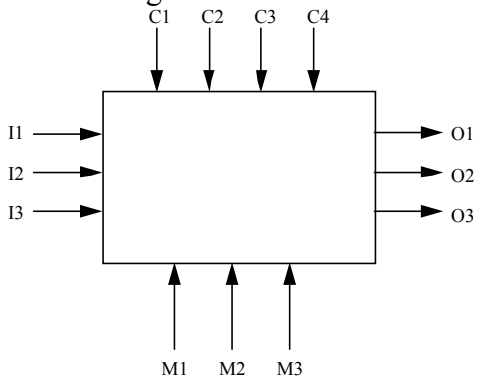
Plusieurs exécutions d'une opération peuvent se réaliser en parallèle.

Les flèches

Chaque flèche correspond à un flux :
- un objet physique,
- une donnée (immatérielle, indépendante de son support)
- un signal (contrôle déclencheur)

dont le nom est indiqué en étiquette

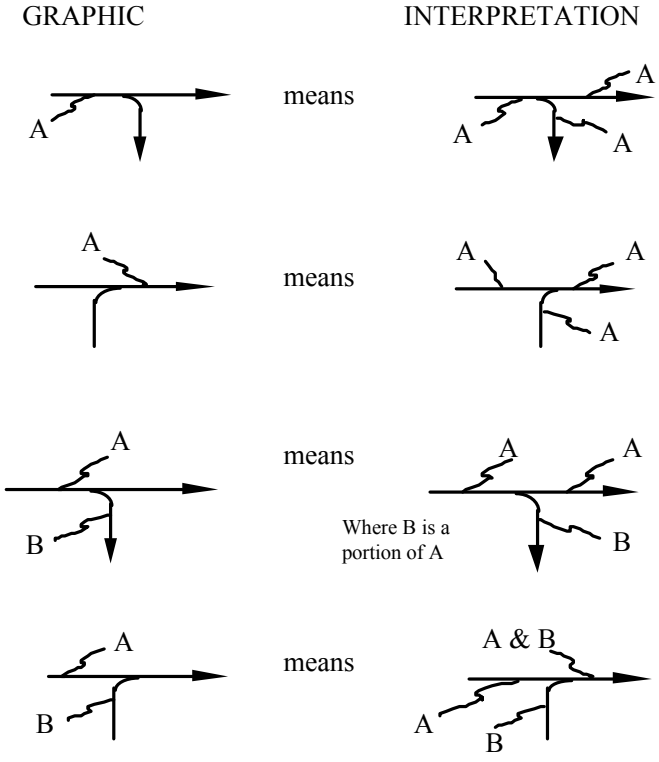
Identification des flèches :
- Par rapport à une boîte ou interface d'un diagramme :
  codification MECS en fonction de son point d'ancrage



- Par rapport à un diagramme :
  - par les boîtes qu'elle connecte : 3to4
  - par son point d'ancrage sur une boîte : 3S2

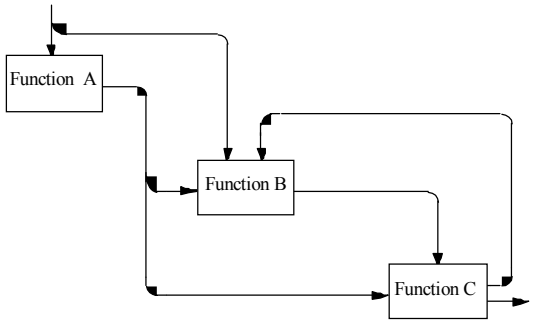
Les flèches correspondent à des câbles, des conduites, qui peuvent
- se ramifier
- se joindre

Conventions :



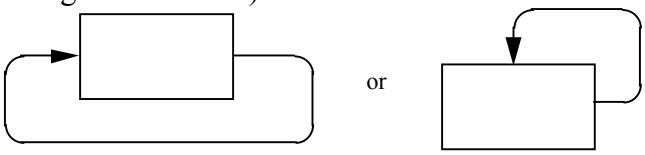
Les relations flèche <—> boîte

L'ancrage des flèches explicite :
pour une boîte : la nature des intrants et des extrants
(tous ne sont pas nécessaires à l'activation de la fonction, OU implicite)
pour une flèches : d'où vient (quelle fonction produit) et où va (quelle fonction utilise) le flux.

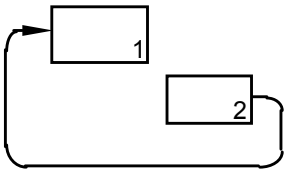


En cas d'indécision entrée / contrôle, privilégier l'aspect contrôle.

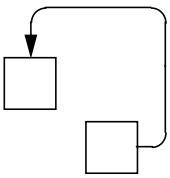
Iterated activity (memory storage or feedback)



Input feedbacks



Control feedbacks



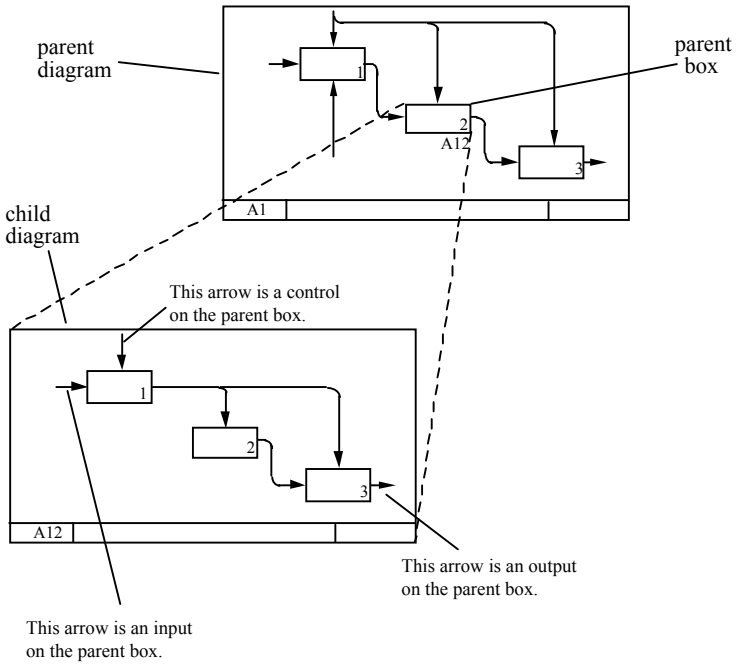
La relation parent – enfant entre diagrammes

Nœud père –>> nœuds fils

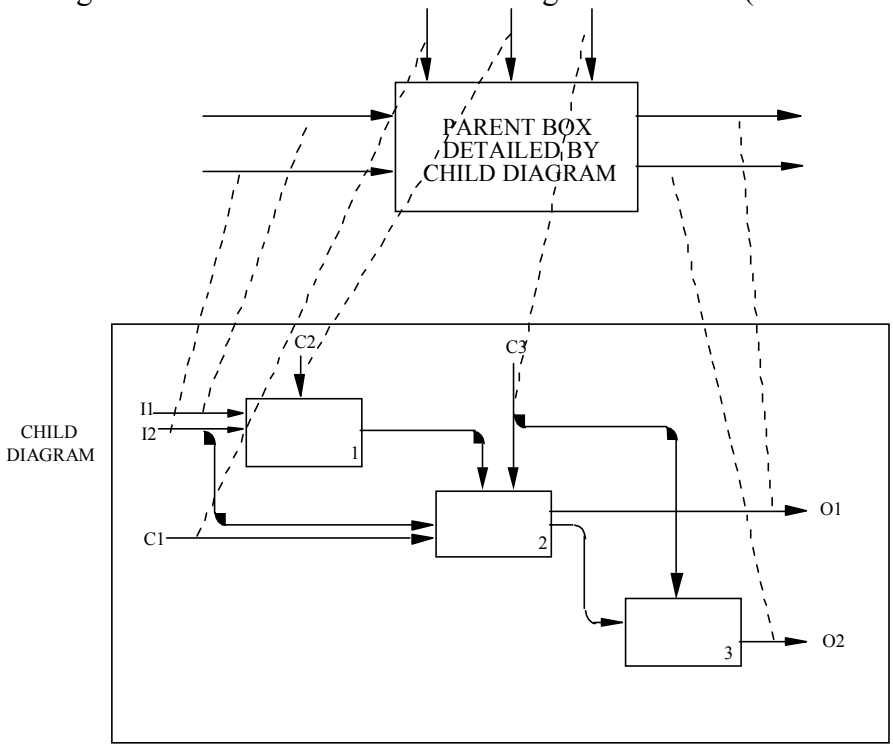
chaque fils détaille l'un des composants du père

Nœud fils –> nœud père

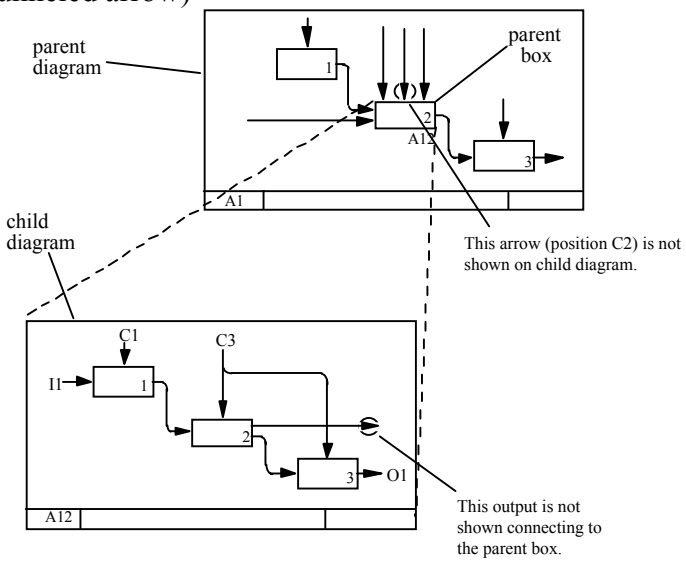
l'interface d'une boite du père (= les flèches connectées) définit le contexte du fils



Convention de nommage des flèches d'interface dans le diagramme enfant (codification MECS)



Flèches masquées (tunneled arrow)



Disposition des boîtes et des flèches

Placer les boîtes selon la règle de dominance

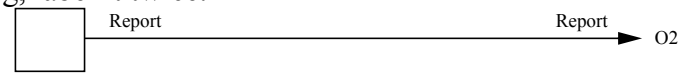
- Les boites importantes doivent être placées le long de la diagonale principale
- Une boîte est dominée par celle placée en haut à gauche

1. Draw arrows along horizontal and vertical lines, not diagonally or as curves (except at corners).

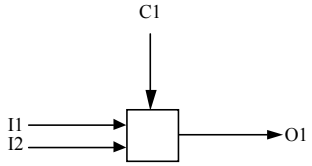
2. Place arrow corners, crossings and labels a reasonable distance away from boxes.

3. Don't use the keywords, i.e., «data», «function», «input», «output», «control"« or «mechanism» in names or labels, unless absolutely necessary.

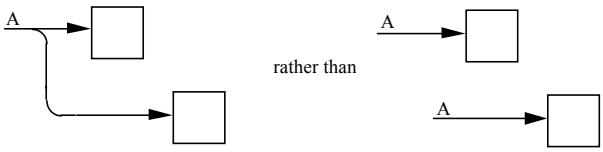
4. If an arrow is long, label it twice.



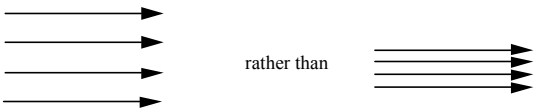
5. Place ICOM codes at the unconnected ends of arrows.



6. Connect open-ended boundary arrows to show all the places affected.
Readers may miss connections otherwise.

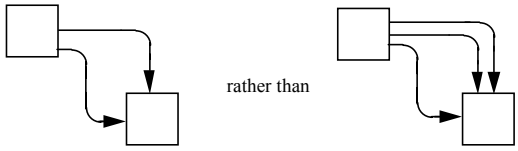


7. Space parallel arrows adequately.
They are hard to follow visually if they are lengthy and close together.

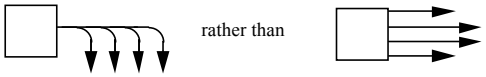


8. Place extra arrowheads along arrows where needed for clarity.

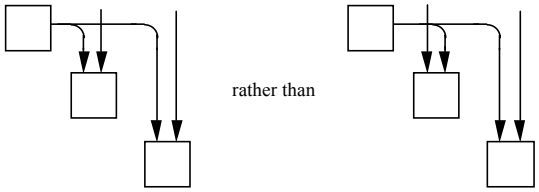
9. Bundle arrows with the same source and the same destination unless the arrow is of such importance that making it part of a pipeline would decrease clarity.



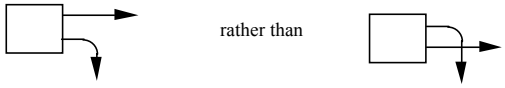
10. Use as few arrows as possible on any one side of a box. If there are too many, bundle some, label with a suitable abstract label and fan out branches to their destinations.



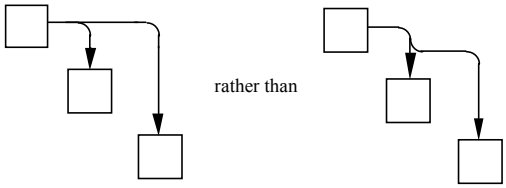
11. If an arrow forks and feeds into several boxes, draw it at the same relative ICOM position on each box, if possible.



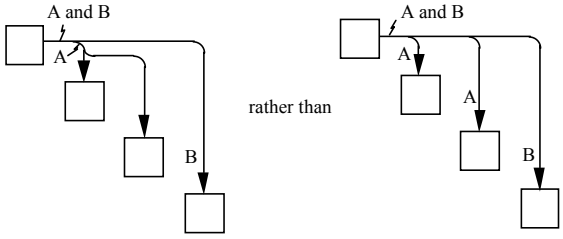
12. Lay out arrows so as to minimize crossings.



13. Minimize curves and corners, while keeping labeled segments clear:



14. Use the expressive potential of branching arrows when and if it is appropriate.



Notation des References

REFERENCE NOTATION	MEANING
2I1	Box 2 Input 1
O2	The boundary arrow with ICOM code O2
2O2 to 3C1 or 2o2 to 3c1	The arrow from 2O2 to 3C1 (The I, O, C or M may be upper case or lower case.)
I2 to 2I3 to 2O2 to (3C1 and 4C2)	From the boundary arrow with ICOM code I2 to Box 2 Input 3, through the activation of Box 2 that yields Output 2, to the availability (via a forking branch) of that output as Control 1 on Box 3 and Control 2 on Box 4.
A21.3C2	On diagram A21 in this model, see Box 3 Control 2..
A42.[3]	On diagram A42, see model note 3.
A42.|3|	Same as above, using optional notation (vertical pipes surrounding model note instead of boxed note).
A42.3	On diagram A42 in this model, see Box 3.
MFG/A42.1	On diagram A42 of the model abbreviated MFG, see Box 1.

Le processus de modélisation SADT

Quelles sont les tâches à réaliser ?
Comment ?
Dans quel ordre ?

1. Préparer le modèle
   répéter
   répéter
   2. Créer un diagramme
   3. Choisir 1 boîte à détailler
   jusqu'à sous-modèle évaluable
   4. Faire un cycle Auteur / Lecteur
   jusqu'à objectif du modèle atteint.

On ne parlera pas de la modélisation des données

1. Préparer le modèle

1.1. Préliminaires

1. Délimiter le contexte du modèle
2. Identifier l'objectif du modèle
3. Choisir un point de vue

Difficiles à formuler d'emblée de façon claire et précise,
améliorer la formulation peu à peu, au fur et à mesure que la compréhension du système s'améliore.

1.2. Recueillir les informations

1. Identifier les sources disponibles d'informations pertinentes
2. Obtenir l'information :
   - interviewer les experts,
   - étudier les documents,

1.3. Etablir la liste des entités

1. Faire une liste des objets, données, messages, informations, …
   qui sont manipulés ou utilisés d'une façon ou d'une autre pour réaliser la fonction
   Prendre en compte les exceptions, les cas particuliers, les annulations.
2. Eliminer les éléments
   hors du contexte,
   dont la prise en compte ne concourt pas à l'objectif
   hors du point de vue
3. Traiter les synonymes

1.4. Structurer les entités

1. Regrouper les éléments qui concourent au même but, véhiculent la même information sous des formes différentes, sont manipulés par les mêmes activités
2. Nommer chaque groupe, selon la relation générique - spécifique,
   On doit pouvoir trouver un nom qui exprime clairement et précisément le contenu de chaque groupe

Objectif : obtenir une liste homogène, dont les éléments se situent au même niveau d'abstraction.

1.5. Définir le Contexte du projet (Créer A-0)

1. Repérer les entités de l'interface, qui supportent les interactions du système avec son environnement
2. Déterminer leur rôle MECS
3. Dessiner la boîte de A-0 avec son interface
4. Rédiger le texte (objectif et point de vue)

Pour établir la liste des entités, il peut être utile de recourir à un diagramme A-1.
On peut aussi commencer directement par A0, et en déduire A-0,
ou revenir sur le A-0 initiale après avoir fait A0.

2. Créer un diagramme

Le Contexte est déterminé par l'interface de la boîte parent que le diagramme détaille
Tout dire sur la boîte parent, ne rien dire d'autre.
Respecter l'objectif et le point de vue de l'ensemble du modèle
On s'arrête généralement au niveau 3, (2 à 30 diagrammes)

2.1. Préparer

Identifier et structurer les éléments du diagrammes avant de le dessiner
Travailler en parallèle sur les entités et les opérations, pour garantir la cohérence du diagramme.

1. Recueillir les informations (cf. 1.2)

2. Etablir une liste des entités (cf. 1.3)

3. Structurer les entités (cf. 1.4)

4. Si on élabore A0 directement (A-0 pourra être déduit de A0),
   repérer les entités qui supportent les interactions du système avec son environnement (cf. 1.5.1).

5. Faire une liste des activités
   - Comment chaque entité est-elle créée, détruite ?
   - Quels sont les états de chaque entité, et quelles activités l'en fait changer ?
   - Pour chaque activité, peut-elle être annulée, corrigée ?
   - il y a-t-il des exceptions ?

Faire une matrice croisée Entités / Activités pour vérifier la complétude des listes
- Chaque Entité est-elle créée, détruite, utilisée ou transformée ?
- Chaque activité a-t-elle une sortie, un contrôle ?

6. Structurer les activités

Regrouper les activités en 3 à 6 groupes, en rassemblant celles qui utilisent les mêmes entités
Les groupes doivent être le plus indépendants possibles :
on doit pouvoir réorganiser la matrice Entités / Activités de telle sorte qu'elle apparaisse comme la composition de petites matrices pleines
On doit pouvoir trouver une forme verbale qui exprime clairement ce en quoi consiste chaque boite

2.2. Dessiner le diagramme

1. placer les boîtes selon la dominances, avec le nom de l'activité
2. tracer les flèches du chemin principal,
   qui va de l'entrée ou contrôle principale à la sortie principale
3. avec le nom des entités, éventuellement complété par leur état
4. attention à la différence entre
   - les objets physiques (unicité de localisation)
   -les données, généralement immatérielles : le nom désigne le contenu et non le support physique.
5. optimiser la disposition graphique
6. tracer les autres flèches d'interface, selon le code MECS
7. tracer les autres flèches intérieures
8. optimiser la disposition graphique

En cas d'incertitude sur le bien fondé d'une flèche, l'écarter.

2.3. Contrôler le diagramme

Vérifications qui portent sur l'intelligibilité, la syntaxe et le contenu du diagramme.

- Taux d'amplification du diagramme :
  apporte suffisamment d'informations nouvelles sans être trop difficile à lire
- Respect de l'objectif et du point de vue
- Respect du Contexte : complétude et pertinence
- Equilibre des niveaux de détail
- Prise en compte de toute et exclusivement l'interface de la boîte parent,
  Respect du code MECS
- Une sortie et un contrôle pour chaque boîte, et moins de 10 flèches
- Clarté du chemin principale de la dominance
- Absence de flèche entre des boîtes proches : pourquoi ?
- Respect des conventions graphiques

2.4. Rédiger les explications

1. Les notes explicatives
2. les diagrammes Pour Explication Seulement
   3. Les Conditions d'Activation de boîtes (cf. Merise)
      <intrants requis> —> <sortie(s) produite(s)>
4. Abonder le Glossaire

3. Choisir la prochaine boîte à détailler

règle 1 :

L'arborescence est construite
- de bas en haut (un enfant après son père)
- en largeur d'abord (les frères avant les enfants)

Avant de décomposer une activité, on s'assure de son interface, et par là de son rôle.
Toute les branches n'ont pas nécessairement la même profondeur.

règle 2 :

Dans une fratrie, choisir la fonction dont la décomposition
- apportera le plus d'information sur les autres boîtes
- la plus difficile, car moins familière ou moins claire

=> la dominante
la moins bien connue, la plus complexe pour l'auteur
généralement pas la plus simple, dont on arrivera tjs à s'accommoder

(Détailler Axj verrouille l'interface des Axk, et donc réduit la marge de manœuvre pour les détailler)

4. Faire un cycle Auteur / Lecteur

Objectifs

- assurer la qualité des diagrammes par la technique de *l'inspection*
- assurer l'homogénéité des composantes du modèle
- favoriser l'esprit d'équipe, l'émergence d'un consensus

de façon efficace

Statut d'un diagramme

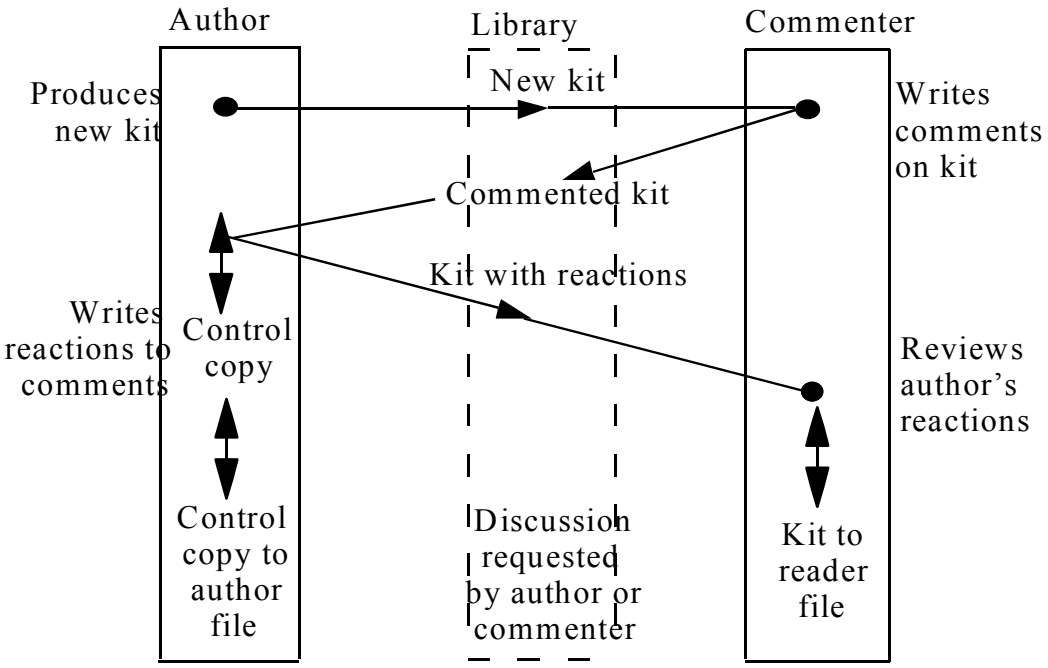
Détermine son degré d'approbation

working (en cours d'élaboration par l'auteur)
draft (engagé dans le cycle auteur / lecteur)
recommended (en attente d'approbation par l'autorité)
publication

Les rôles des participants

Détermine la responsabilité de chaque participant vis à vis d'un diagramme

Expert : connaisseur du domaine, donne les informations, valide les diagrammes
Auteur : élabore le diagramme
Lecteur (Examinateur, Commentateur) : relit le diagramme, partage la responsabilité de sa qualité
Bibliothécaire : assure la communication des documents, et garde la trace de leur circulation.
Autorité : arbitre les désaccords, décide de la publication d'un diagramme



4.1. Les étapes d'un cycle

Dans la version originale, toutes les fonctions relatives à
la transmission des documents,
l'enregistrement de leurs versions successives
sont assurées par un bibliothécaire.

Produire un kit
rassembler un ensemble cohérent de documents (4-5 diagrammes avec leur texte, glossaire, PES)
l'identifier
le diffuser aux lecteurs avec le délai de réponse souhaité (1 – 3 jours)

Commenter le kit
Rédiger sur la copie du kit des notes de lecteur, en rouge
Chaque note porte sur un diagramme ouX l'ensemble du kit
Etre positif

Répondre aux commentaires
Lire tous les commentaires avant de réagir
Pour chaque note,
- V accord : modification du diagramme –> nouvelle version
- X désaccord : justification par une note en réponse à celle du lecteur, en bleu

Examiner les réponses
accord avec les réponses (et la nouvelle version) : classement du kit
désaccord :
un nouveau cycle (pas plus de 2 ou 3 pour un kit)
entretien auteur – lecteur
réunion avec l'autorité
L'auteur peut ne pas prendre en compte un commentaire d'un lecteur
la responsabilité de ce dernier est alors dégagée

4.2. Lire un diagramme

Comment aborder un diagramme pour comprendre rapidement ce qu'il contient ?

1. Lire la fonction des boîtes, en diagonale
2. Examiner la boîte parent, et identifier les flèches les plus importantes
3. Chercher le chemin principale, qui relie l'entrée (ou le contrôle) la + importante à la sortie principale
4. Examiner chaque boîte dans l'ordre pour vérifier que chaque intrant se justifie
5. Examiner les autres flèches et chercher les autres chemins, les contre-réactions, les erreurs
6. Lire les textes

Comparer un diagramme avec une version antérieure est difficile.

4.3. Commenter un diagramme

Le diagramme est-il
- correcte syntaxiquement ?
- compréhensible pour le lecteur ?
- conforme à sa compréhension des choses ?

Cf. 2.3 Contrôler le diagramme

Caractéristiques d'une note :
- une idée –> une note
- concise
- claire, précise sur la nature du désaccord
- suggérer une solution

Faire des notes de synthèse sur l'ensemble du diagramme ou du kit pour identifier la cause du désaccord

4.4. Améliorer un diagramme

1. Décomposer une boîte en plusieurs
2. Regrouper plusieurs boîtes en une seule
3. Ajouter, retirer, regrouper des interfaces à une boîte
   => ajouter, ramifier, joindre une flèche.

Dès que l'on modifie l'interface d'un diagramme,
les conséquences sur la boîte mère et les diagrammes qui détaillent ses sœurs peuvent être lourdes.

Eviter
- l'instabilité
- les rustines (modification a minima qui altèrent la cohérence du modèle).

IDEF COVER SHEET FORM

MODEL/DOCUMENT DESCRIPTION		PROJECT INFORMATION		KIT INFORMATION	REVIEW CYCLE
TITLE:		AUTHOR:	DATE:	STANDARD KIT	REVIEWER DATE
LIFE CYCLE STEP:				SUMMARY KIT	
IDEF METHOD:	SYSTEM:	COMPANY:	TASK NO.	SUPERSEDED OR REVISED DOCUMENT NUMBER	AUTHOR DATE
DISTRIBUTION TYPE:					

COPY FOR COMMENT	COPY FOR READ	REVIEWERS NAME	COMPANY	PROJECT NUMBER

COPY FOR COMMENT	COPY FOR READ	REVIEWERS NAME	COMPANY	PROJECT NUMBER

	LOG
	FILE
	AUTHOR

KIT CYCLE DATES

RECEIVED BYLIBRARY	
KIT TO REVIEWER	
COMMENTS DUE BACK TO LIBRARY	
COMMENTS TO AUTHOR	
AUTHOR RESPONSE DUE BACK TO LIBRARY	
AUTHOR RESPONSE TO COMMENTER	
KIT CYCLE COMPLETE	

COPYING INSTRUCTIONS

____ copies of ____ pages = ____ total

NODE INDEX/CONTENTS

Pg.	Node	Title	C-Number	Status

COMMENTS/SPECIAL INSTRUCTIONS

C-NUMBER	PAGE

NOMENCLATURE	DOCUMENT/MODEL TITLE	DOCUMENT NUMBER
		1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

IDEF KIT DIAGRAM FORM

Used at:	Author:	Date:		WORKING	READER	DATE	Context:
	Project:	Rev:		DRAFT			
	Notes: 1 2 3 4 5 6 7 8 9 10			RECOMMENDED			
				PUBLICATION			

Node:	Title:	Number:
		Page: